Csendes, T. and Ratz, D.

A review of subdivision direction selection in interval methods for global optimization

This paper gives a short overview of the latest results on the role of the interval subdivision selection rule in branch-and-bound algorithms for global optimization. The class of rules that allow convergence for two slightly different model algorithms is characterized, and it is shown that the four rules investigated satisfy the conditions of convergence. An extensive numerical study with a wide spectrum of test problems indicates that there are substantial differences between the rules in terms of the required CPU time, the number of function and derivative evaluations and space complexity. Two of the rules can provide substantial improvements in efficiency.

1. Introduction

Interval subdivision methods for global optimization [4, 8] aim at providing reliable solutions to global optimization problems

$$\min_{x \in X} f(x), \tag{1}$$

where the objective function $f : \mathbb{R}^n \to \mathbb{R}$ is continuously differentiable, and the search region $X \subseteq \mathbb{R}^n$ is an n-dimensional interval. No special problem structure is required: only inclusion functions of the objective function and its gradient are utilized [1]. Denote the set of compact intervals by $\mathbb{I} := \{[a, b] \mid a \leq b,\ a, b \in \mathbb{R}\}$ and the set of n-dimensional intervals (also called simply intervals or boxes) by $\mathbb{I}^n$. We call a function $F : \mathbb{I}^n \to \mathbb{I}$ an *inclusion function* of $f : \mathbb{R}^n \to \mathbb{R}$ in X, if $x \in Y$ implies $f(x) \in F(Y)$ for each interval Y in X. In other words, $f(X) \subseteq F(X)$, where $f(X)$ is the range of $f(x)$ on X. The inclusion function of the gradient of $f(x)$ is denoted by $F'(X)$.

There are several ways to build an inclusion function for a given optimization problem (e.g. by using the Lipschitz constant). Interval arithmetic [1] is a convenient tool for constructing the inclusion functions, and one can get inclusion functions for almost all functions that can be calculated by a finite algorithm (i.e. not only for given expressions).

It is assumed in the following that the inclusion functions have the *isotonicity* property, i.e. $X \subseteq Y$ implies $F(X) \subseteq F(Y)$, and that for all the inclusion functions $w(F(X^i)) \to 0$ as $w(X^i) \to 0$ holds, where $w(X)$ is the width of the interval X ($w(X) = \max X - \min X$ if $X \in \mathbb{I}$, and $w(X) = \max_{i=1}^n w(X_i)$, if $X \in \mathbb{I}^n$).

2. Subdivision direction selection rules in branch-and-bound algorithms

We investigate the following model algorithm that comprises the most important common features of interval methods for global optimization (cf. e.g. [4, 8]).

Step 0 Set $Y = X$ and $y = \min F(X)$. Initialize the list $L = ((Y, y))$ and the cut-off level $z = \max F(X)$.

Step 1 Choose a coordinate direction $k \in \{1, 2, \ldots, n\}$.

Step 2 Bisect Y in direction k: $Y = V^1 \cup V^2$.

Step 3 Calculate $F(V^1)$ and $F(V^2)$, and set $v^i = \min F(V^i)$ for $i = 1, 2$ and $z = \min\{z, \max F(V^1), \max F(V^2)\}$.

Step 4 Remove (Y, y) from the list L.

Step 5 Cut-off test: discard the pair (V^i, v^i) if $v^i > z$ (where $i \in \{1, 2\}$).

Step 6 Monotonicity test: discard any remaining pair (V^i, v^i) if $0 \notin F'_j(V^i)$ for any $j \in \{1, 2, \ldots, n\}$, and $i = 1, 2$.

Step 7 Add any remaining pair(s) to the list L. If the list becomes empty, then STOP.

Step 8 Denote the pair with the smallest second element by (Y, y).

Step 9 If the width of $F(Y)$ is less than ε, then print $F(Y)$ and Y, STOP.

Step 10 Go to Step 1.

The interval Y that is first set in Step 0, and then updated in Step 8, is called the *leading box*, and the leading box of the iteration number s is denoted by Y^s. Notice that the cut-off test does not have any effect on the convergence of the algorithm; it may just decrease the space complexity, the maximal length of list L.

The interval subdivision direction selection rule in Step 1 is the target of our present study. All the rules select a direction with a merit function:

$$k := \min\left\{j \mid j \in \{1, 2, \ldots, n\} \text{ and } D(j) = \max_{i=1}^{n} D(i)\right\} \tag{2}$$

where $D(i)$ is determined by the given rule. For Rule A $D(i) := w(X_i)$ (see e.g. [8]). Rule B selects the coordinate direction, for which (2) holds with $D(i) := w(F_i'(X))w(X_i)$ (c.f. [4]). For Rule C the merit function is $D(i) := w(F_i'(X)(X_i - m(X_i)))$ ([6, 9]). Rule D is derivative-free like Rule A, and reflects the machine representation of the inclusion function $F(X)$ (see [9]): $D(i) := w(X_i)$ if $0 \in X_i$, and $w(X_i)/\min\{|x_i| \mid x_i \in X_i\}$ otherwise. Such algorithmic improvements can be quite important for some real life applications (like e.g. [7]).

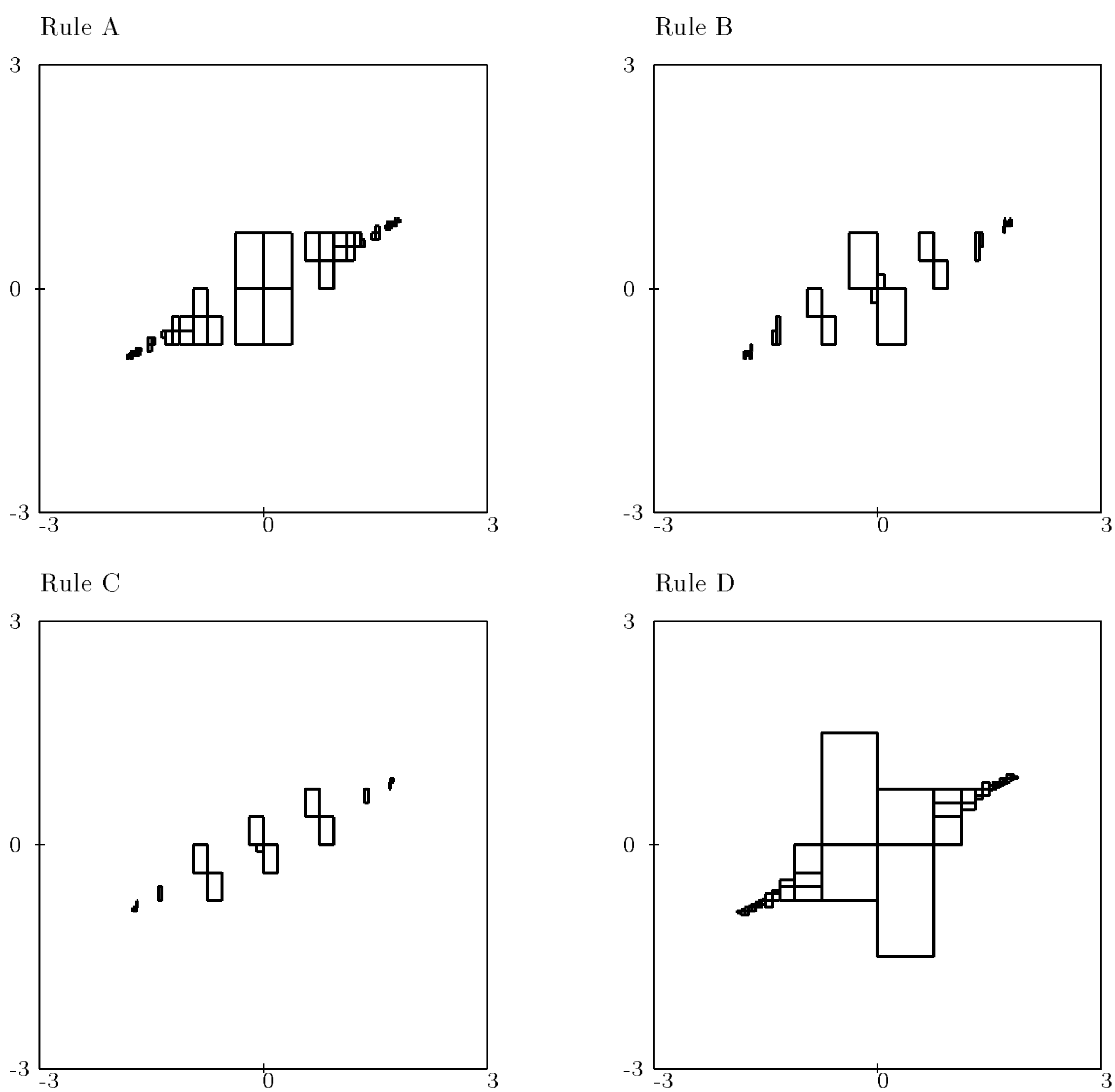

Figure 1: Remaining subintervals after 250 iteration steps of the model algorithm with the direction selection rules A, B, C, and D for the Three-Hump-Camel-Back problem.

Typical distributions of subintervals are shown in Figure 1 for the direction selection rules A, B, C, and D, respectively. The direction selection rule A tends to form square-like boxes, while Rule D produces elongated intervals as the magnitudes of the coordinates differ. Rules B and C generate similar sets of subintervals, reflecting the utilized derivative information. The remaining number of subboxes were 53, 33, 19 and 52 , respectively.

3. Direction selection rules and the convergence

In [3] and [10], the relation of subdivision direction selection rules to convergence has been studied. We can just highlight here the most important results. First a property (balanced) of the subdivision direction selection rules was defined. A similar property was discussed in [5]. If an otherwise arbitrary selection rule has this property, than the related model algorithm converges both in the sense that the width of the inclusion function of the objective function converges to the global minimum ($\lim_{s\to\infty} F(Y^s) = f^*$) and also in the sense that the width of the leading boxes goes to zero ($\lim_{s\to\infty} w(Y^s) = 0$). The opposite direction also holds with some unimportant exceptions: once the model algorithm converges in the first sense, then the direction selection rule must be balanced. If the model algorithm converges for a problem in the second sense, then either the direction selection rule is balanced, or the given problem has a positive width subinterval full with global minimizer points — that is located with the procedure.

The subdivision direction selection rules A and D are balanced. The model algorithm with Rules B and C either converges to a positive width interval that contains exclusively global minimizer points, or the subdivision rule acts as a balanced rule. Thus, in the overwhelming majority of cases, all the four investigated Rules work like a balanced rule, and hence the model algorithm converges in both senses. For the exceptional case when a positive width interval exists that contains exclusively global minimizer points, the provided solution is even more valuable than a single point solution would be.

The first mentioned paper [3] studied the general case, while the second one [10] investigated a slightly modified algorithm. In Step 8 of the latter, the oldest one of the list members with the smallest second element y is selected for further processing. While the original algorithm converges to a single global minimizer point, the modified one converges to all non-hidden global minimizers. Paper [10] showed that the theoretical results regarding the effects of subdivision direction selection rules on the convergence are valid for the modified algorithms too.

4. Numerical experiences

The numerical tests were carried out on large problem set [3, 10] containing all the standard test problems, all the problems studied in [4], and many more. The general algorithm was coded in FORTRAN-90, and used natural interval extension and handcoded gradients, while the modified algorithm was coded in PASCAL-XSC and applied the built-in interval data types and automatic differentiation.

Effort	Rule B			Rule C			Rule D		
measure	best	average	worst	best	average	worst	best	average	worst
CPU	8%	93%	145%	6%	94%	155%	19%	943%	32 620%
NFE	20%	93%	197%	18%	93%	195%	20%	202%	3 720%
NDE	20%	93%	205%	18%	93%	203%	19%	198%	3 572%
list length	20%	101%	304%	19%	103%	296%	19%	234%	5 041%

Table 1: Relative effort values of the general model algorithm compared to those obtained with Rule A.

While the detailed numerical results can be found in [3], Table 1 contains the most important comparison terms regarding the CPU time, number of objective function (NFE) and derivative (NDE) evaluations, and the list length necessary to solve the given problems. According to these figures, Rules B and C are slightly better than Rule A, and Rule D proved to be the worst of the four studied rules. For the hard-to-solve problems the mentioned differences were even sharper.

Effort measure	Rule B best	Rule B average	Rule B worst	Rule C best	Rule C average	Rule C worst	Rule D best	Rule D average	Rule D worst
CPU	6%	84%	130%	6%	78%	130%	19%	119%	353%
NFE	7%	85%	119%	7%	81%	119%	18%	119%	323%
NDE	6%	86%	121%	6%	81%	121%	18%	118%	338%
list length	5%	89%	117%	5%	86%	129%	14%	116%	283%

Table 2: Relative effort values of the modified model algorithm compared to those obtained with Rule A.

The detailed numerical results for the modified algorithm are discussed in [10]. Table 2 comprises the relative effort measures for the modified procedure. These figures reflect the same consequences as drawn on the basis of Table 1 — with some alterations in the differences.

In a recent study [2], the model algorithm was completed with many sophisticated accelerating devices (such as the optimal centered form for the inclusion function, interval Newton steps, nonconvexity test and multisection strategies). The above consequences regarding the role of the direction selection rules were confirmed by this algorithm — with obvious slight differences in the actual efficiency improvement figures.

5. Summary and conclusions

All the four studied interval subdivision selection rules allow the convergence of the model algorithm. According to the available numerical tests, Rule B and Rule C can decrease the computational and space complexity of the majority of global optimization problems. These improvements are especially large for hard-to-solve problems.

In spite of some efforts made, it is still open how one can achieve similar efficient solutions when the gradient information cannot be used in the direction selection rule (like in Rules A and D). A rule utilizing second order derivative information in a clever way is also missing.

Acknowledgements

The work has been supported by the Grants OTKA T 016413, T 017241, MKM 414/1994, and KDAW 960.4-90.

6. References

1 Alefeld, G., Herzberger, J.: Introduction to Interval Computations; Academic Press, New York 1983.
2 Berner, S.: Ein paralleles Verfahren zur verifizierten globalen Optimierung, PhD thesis, submitted to Fachbereich Mathematik, Bergische Universität GH Wuppertal, 1995.
3 Csendes, T., Ratz, D.: Subdivision direction selection in interval methods for global optimization; to appear in SIAM Journal of Numerical Analysis.
4 Hansen, E.: Global Optimization Using Interval Analysis; Marcel Dekker, New York 1992.
5 Horst, R., Nast, M., Thoai, N.V.: A New LP-Bound in Multivariate Lipschitz Optimization: Application to Unconstrained and Linearly Constrained Problems and to Systems of Inequalities, Forschungsbericht Nr. 94-08, Universität Trier, 1994.
6 Kearfott, R.B.: A FORTRAN-90 Environment for Research and Prototyping of Enclosure Algorithms for Constrained and Unconstrained Nonlinear Equations, ACM T. on Mathematical Software, **21** (1995), 63–78.
7 Kristinsdottir, B.P., Zabinsky, Z.B., Csendes, T., Tuttle, M.E.: Methodologies for tolerance intervals, Interval Computations, **3** (1993), 133–147.
8 Ratschek, H., Rokne, J.: New Computer Methods for Global Optimization; Ellis Horwood, Chichester 1988.
9 Ratz, D.: Automatische Ergebnisverifikation bei globalen Optimierungsproblemen. PhD Thesis, Universität Karlsruhe, 1992.
10 Ratz, D., Csendes, T.: On the Selection of Subdivision Directions in Interval Branch-and-Bound Methods for Global Optimization; to appear in the J. of Global Optimization.

Addresses: Tibor Csendes Bergische Universität GH Wuppertal, Fachbereich Mathematik, Gaußstr. 20, D-42097 Wuppertal, Germany (on leave from the Institute of Informatics, József Attila University, Szeged, Hungary), email: csendes@inf.u-szeged.hu.
Dietmar Ratz Universität Karlsruhe, Institut für Angewandte Mathematik, Kaiserstr. 12, D-76128 Germany, email: Dietmar.Ratz@math.uni-karlsruhe.de.